REMARKS

Claims 1-52 are pending in the application, of which Claims 1, 11, 13, 32, 33, 41, 46 and 50-52 are independent claims. Claims 1, 7-9, 11-13 and 14-52 were rejected under 35 U.S.C. § 102(e). Claims 2-6 and 10 were rejected under 35 U.S.C. § 103(a). Claims 10, 11, 13, 41, 46 and 50-52 are amended to claim the invention more distinctly. For the reasons stated below, it is believed that all claims are now in condition for allowance.

Rejections under 35 U.S.C. § 102(e)

Claims 1, 7-9, 11-13 and 14-52 are rejected under 35 U.S.C. § 102(e) based on U.S. Patent No. 6,092,213 to Lennie. This rejection is traversed. For the convenience of the Examiner, the Applicants will first address the rejections to independent Claims 1, 11, 13, 46 and 50-52, which relate to, among other things, a member node requesting a change to the cluster definition by sending a proposed change to the shared repository. The Applicants will then address the rejections to Claims 33, 41, 50 and 52, which relate to, among other things, a potential member node accessing the cluster definition on the shared repository.

In disclosed embodiments of the invention, a cluster definition for a network cluster is maintained in a shared repository. A single node is selected as the coordinator of the cluster definition is the only node that is authorized to make updates to the definition. Each node has a scratch area on the shared repository that it uses to request changes to the cluster definition. The coordinator node applies the requested changes to the shared repository. When a potential member node wishes to join the cluster, it accesses the current cluster definition on the shared repository before needing to establish network connectivity with the cluster.

By contrast, Lennie is directed a technique for distributing "configuration data to each node so that each node has a database containing the configuration data associated with that node."[1] A primary process is responsible for receiving requests that require modification of the configuration data. The primary process sends the requested change to a master audit log and then distributes the requested change to all the nodes. Once the primary process has distributed

---

[1] See Lennie, Abstract.

the change to all the nodes, it updates the master audit log to note that the requested change has been effected throughout the system.

A.   Rejections of Claims 1, 11, 13, 46 and 50-52

Claim 1 requires that a member node <u>request</u> a change to the cluster definition by <u>sending</u> a <u>proposed</u> change to a shared repository. Claims 11, 13, 46 and 50-52 have been amended accordingly. By having member nodes send their proposed changes to a shared repository, the invention can avoid a situation in which multiple nodes try to make changes to the cluster definition in parallel.[2] Parallel edits can result in a cluster definition which partially represents the changes made by a first node and partially represents changes made by a second node. Thus, the claimed invention can avoid this situation by allowing nodes to submit their proposed changes to the cluster definition on the shared repository.

In comparison, Lennie is silent as to how a member node can propose a change to the cluster definition. In particular, Lennie does not relate to a technique for requesting changes to the cluster definition. Lennie explains that requests to change the definition are received, and thus, implemented via the primary process. Specifically, in Lennie, the primary process receives the requested change, logs the requested change and propagates the requested change to all the nodes. Thus, Lennie is not directed to the claimed **member node requesting a change by sending a proposed change to the shared repository**, and, therefore, Lennie does not discuss the limitations and advantages of the claimed invention.

To establish a prima facie case for anticipation under 35 U.S.C. § 102(b), the cited reference must teach every aspect of the claimed invention either explicitly or impliedly. Because Lennie does not teach the claimed member node requesting a change to the cluster definition by sending a proposed change to the shared repository, it does not teach every aspect of Claims 1, 11, 13, 46 and 50-52.

---

[2] See Specification, pg. 14, ll. 25 - pg. 15, ll. 2.

In addition, addressing Claim 51, in particular, this claim specifically requires that a member node request a change to the cluster definition by:

> sending a proposed change to a scratch area;
> setting a valid bit associated with the scratch area;
> verifying the valid bit;
> setting an update flag;
> modifying the cluster definition to reflect the requested change; and
> logging a progress of modifying the cluster definition in a log file in
> parallel with modifying the cluster definition;
> incrementing a version number associated with the shared repository; and
> clearing the valid bit and the update flag.

In comparison, Lennie does not discuss anything about member nodes requesting changes by sending a proposed change to a scratch area of the repository. As discussed above, Lennie does not even discuss how a member node can <u>request</u> a change to the cluster definition. Thus, Lennie does not address the problems associated with member nodes requesting definition changes, or suggest the solutions presented in Claim 51.

B.   Rejections of Claims 33, 41, 50 and 52

In Lennie, the primary process communicates the cluster definition to each node, and each node has a monitor process that stores the definition in the node's corresponding database registry. Thus, in Lennie, <u>network connectivity is necessary</u> for a node to receive changes to the cluster definition.

By contrast, Claims 33, 41, 50 and 52 require that a member be coupled to a shared repository (which stores the cluster definition), a coordinator node to update the definition in the shared repository, and a potential member node to access the cluster definition on the shared repository. Thus, the member node, coordinator node and potential member node, all access the same shared repository to read the cluster definition. Further, with this technique, a potential member node only needs access to the shared repository to read the cluster definition. Unlike Lennie, the claimed potential member node can access this definition on the shared repository before establishing network connectivity with the cluster.

In addition, Lennie does not discuss anything about a potential member node. In Lennie, all nodes described are already members of the network cluster. Lennie relates to distributing cluster definition changes to member nodes with a primary process and does not consider nodes that are not members. Thus, Lennie does not describe how a potential member can obtain the cluster definition, and therefore does not discuss the limitations and advantages of the claimed invention.

Because Lennie does not teach the claimed potential member node accessing the cluster definition on the shared repository, it does not teach every aspect of Claims 33, 41, 50 and 52.

Addressing Independent Claim 52, in particular, this claim requires that a potential member node request membership in the network cluster, and access the cluster definition on the shared repository including:

> determining a version number of the shared repository to yield a first version number;
>
> reading the cluster definition;
>
> re-determining a version number of the shared repository to yield a second version number;
>
> comparing the first version number with the second version number; and
>
> repeating the step of accessing the cluster definition until the first version number equals the second version number.

Lennie, however, does not discuss these claimed limitations. Lennie does not relate to member nodes requesting membership in the cluster. Thus, Lennie does not teach every aspect of Claim 52.

Based on the above remarks, it is respectfully requested that the rejections to Independent Claims 1, 11, 13, 32, 33, 41, 46 and 50-52 under 35 U.S.C. § 102(e) be withdrawn.

Claims 7 and 9 depend from base Claim 1, Claims 12 and 23-31 depend from base Claim 11, Claims 14-22 depend from base Claim 13, Claims 34-40 depend from base Claim 33, Claims 42-45 depend from base Claim 41 and Claims 47-49 depend from base Claim 46. Because elements of Independent Claims 1, 11, 13, 32, 33, 41, 46 have been shown above to be absent

from the teachings of Lennie, either explicitly or impliedly, then each of the Dependent Claims 7, 9, 12, 14-22, 23-31, 34-40, 42-45 and Claims 47-49 are also not anticipated by Lennie. Therefore, it is respectfully requested that the rejections to Claims 7, 9, 12, 14-22, 23-31, 34-40, 42-45 and Claims 47-49 under 35 U.S.C. § 102(e) be withdrawn

Rejections under 35 U.S.C. § 103(a)

Dependent Claims 2-6 and 10 depend from base Claim 1, and have been rejected under 35 U.S.C. § 103(a). Specifically, Dependent Claims 2, 3 and 10 are rejected under 35 U.S.C. § 103(a) based on U.S. Patent No. 6,092,213 to Lennie in view of U.S. Patent No. 6, 014,669 to Slaughter. Dependent Claim 4 is rejected under 35 U.S.C. § 103(a) based on Lennie in view of U.S. Patent No. 6,003,075 to Arendt. Dependent Claim 5 is rejected under 35 U.S.C. § 103(a) based on U.S. Patent No. 6,092,213 to Lennie in view of U.S. Patent No. 5,964,886 to Slaughter. Dependent Claim 6 is rejected under 35 U.S.C. § 103(a) based on Lennie and Slaughter in view of U.S. Patent No. 6,243,702 to Bramford.

As the dependent claims incorporate all limitations from their corresponding base claims, allowance of the dependent claims follows from allowance of the base claim. Because the base claim are in condition for allowance, the dependent claims should also be allowed. Therefore, it is respectfully requested that the rejections to Dependent Claims 2-6 and 10 under 35 U.S.C. § 103(a) be withdrawn.
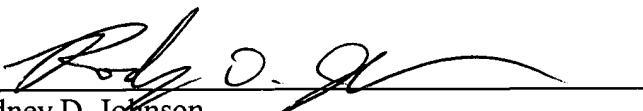
Claim Amendments

Claims 10, 11, 13, 41, 46 and 50-52 are amended to claim the invention more distinctly. These amendments are not in acquiescence to the rejections. In particular, Dependent Claim 10 is amended to correct a typographical error. Independent Claims 11, 13, 46, 50 and 52 are amended to state that a member node requests a change to the cluster definition by sending a proposed change to the shared repository, as similarly recited in Independent Claims 1, 32 and 51. Claim 41 is amended to specify that a potential member node accesses the cluster definition on the shared repository. Claim 51 is amended to clarify that a member node requesting a change to cluster definition by sending. . ., setting. . ., verifying. . ., among others. Support for these amendments can be found in the originally filed Claims 1-13, and Specification, for example, pg. 4. Thus, no new matter is being introduced. Acceptance is respectfully requested.

CONCLUSION

In view of the above amendments and remarks, it is believed that all claims are in condition for allowance, and it is respectfully requested that the application be passed to issue. If the Examiner feels that a telephone conference would expedite prosecution of this case, the Examiner is invited to call the undersigned attorney at (978) 341-0036.

Respectfully submitted,

HAMILTON, BROOK, SMITH & REYNOLDS, P.C.

By _____

Rodney D. Johnson
Registration No.36,558
Telephone: (978) 341-0036
Facsimile: (978) 341-0136

Concord, MA 01742-9133
Dated: June 12, 2003

MARKED UP VERSION OF AMENDMENTS

<u>Claim Amendments Under 37 C.F.R. § 1.121(c)(1)(ii)</u>
Please amend Claims 10, 11, 13, 41, 46 and 50-52.

10.   (Amended) The method of Claim 2 further including the step of:

re-requesting, [bu] <u>by</u> the member node, the change to the cluster definition if after a period of time, the change is not made to the cluster definition.

11.   (Twice Amended) An apparatus for updating a cluster definition for a network cluster having at least one member node, comprising:

a shared repository coupled to the at least one member node of the cluster, the repository including the cluster definition<u>;</u> [and a proposed change to the cluster definition; and]

<u>a member node to request a change to the cluster definition by sending a proposed change to the shared repository; and</u>

a coordinator node, selected from the at least one member node of the network cluster, to update the cluster definition with the proposed change.

13.   (Twice Amended) A computer program product for maintaining a cluster definition for a network cluster having at least one member node, the computer program product comprising:

a computer usable medium having computer readable program code thereon, including program code for:

coupling the at least one member node to a shared repository;

storing a cluster definition for the network cluster in the shared repository;

selecting a coordinator node from the at least one member node of the network cluster; [and]

<u>requesting a change to the cluster definition by sending a proposed change to the shared repository; and</u>

directing the coordinator node to update the cluster definition in response to [a] <u>the</u> request<u>ed</u> [to] change <u>to</u> the cluster definition.

41.   (Amended) An apparatus for maintaining a cluster definition for a network cluster having at least one member node, comprising:

a shared repository coupled to the at least one member node of the cluster, the shared repository including the cluster definition and a proposed change to the cluster definition;

a coordinator node, selected from the at least one member node of the network cluster, to update the cluster definition with the proposed change; and

a potential member node to access the cluster definition <u>on the shared repository</u>.

46.   (Amended) A computer program product for maintaining a cluster definition for a network cluster having at least one member node, the computer program product comprising:

a computer usable medium having computer readable program instructions thereon, including instructions for:

coupling the at least one member node to a shared repository;

storing a cluster definition for the network cluster in the shared repository;

selecting a coordinator node from the at least one member node of the network cluster;

<u>requesting a change to the cluster definition by sending a proposed change to the shared repository;</u>

directing the coordinator node to update the cluster definition to reflect [a] <u>the</u> requested change; and

directing a potential member node to access the cluster definition <u>on the shared repository</u>.

50.   (Amended) A system for maintaining a cluster definition for a network cluster having at least
one member node, the system comprising:

      a means for coupling the at least one member node to a shared repository;

      a means for storing a cluster definition for the network cluster in the shared
repository;

      a means for selecting a coordinator node from the at least one member node of
the network cluster;

      a means for requesting a change to the cluster definition by sending a
proposed change to the shared repository;

      a means for the coordinator node to update the cluster definition to reflect the
requested change; and

      a means for a potential member node to access the cluster definition on the
shared repository.

51.   (Amended) A method for maintaining a cluster definition for a network cluster having at least
one member node, the method comprising:

      coupling the at least one member node to a shared repository;

      storing a cluster definition for the network cluster in the shared repository;

      selecting a coordinator node from the at least one member node of the network
cluster;

      at a member node, requesting a change to the cluster definition by[; for each requested
change]:

            sending a proposed change to a scratch area of the shared repository;

            setting a valid bit associated with the scratch area;

            verifying the valid bit;

            setting an update flag;

            modifying the cluster definition to reflect the requested change; and

            logging a progress of modifying the cluster definition in a log file in parallel
with modifying the cluster definition;

            incrementing a version number associated with the shared repository; and

            clearing the valid bit and the update flag; and

      from the coordinator node, updating the cluster definition to reflect the requested
change.

52. (Amended) A method for maintaining a cluster definition for a network cluster having at least one member node, the method comprising:

coupling the at least one member node to a shared repository;

storing a cluster definition for the network cluster in the shared repository;

selecting a coordinator node from the at least one member node of the network cluster;

at a member node, requesting a change to the cluster definition <u>by sending a proposed change to the shared repository</u>;

from the coordinator node, updating the cluster definition to reflect the requested change;

requesting, by a potential member node, membership in the network cluster; and

accessing, by the potential member node, the cluster definition <u>on the shared repository</u>, for each potential member node accessing the cluster definition:

determining a version number of the shared repository to yield a first version number;

reading the cluster definition;

re-determining a version number of the shared repository to yield a second version number;

comparing the first version number with the second version number; and

repeating the step of accessing the cluster definition until the first-version number equals the second version number.